BSEEMP manual

Ataru Tanikawa

Contents

1 Change log

- 3rd March 2022 This manual created

2 Overview

We describe the manual of BSEEMP, a binary population synthesis code. The BSEEMP code is based on the BSE code (Hurley et al., 2000, 2002). The BSEEMP code extends the functionality of the original BSE code to more metal-poor stars and more massive stars. The BSEEMP code changes binary evolution models, keeping backward compatibility.

3 Getting started

We describe the first step to use the BSEEMP code. As an example, we show how to use the BSEEMP to follow evolution of a binary star with the M model (see section 4 if you want to know what the M model is). If you want to follow evolution of a large number of binary stars, or evolution of a binary star with the L model, please see section 5.

3.1 Environment

The BSEEMP code works on Linux (maybe on Mac OS X and Windows with WSL or cygwin). To compile the BSEEMP, you need the "make" command, and fortran and C++ compilers. We tested GCC compiler, and expect ICC compiler also works well.

3.2 Install

You can get the BSEEMP in the following ways:

- Using browsers
  1. Click "Download ZIP" in `https://github.com/atrtnkw/bseemp` to download bseemp-main.zip
  2. Move the zip file to the directory under which you want to install the BSEEMP code and unzip the file.
- Using command line interface
  1. execute the command "`git clone git@github.com:atrtnkw/bseemp.git`"

3.3 Compiling and running

You can compile an excutable file of BSEEMP in the following way:

1. Move to the directory `$(BSEEMP)/src`, where `$(BSEEMP)` denotes the hightest-level directory of BSEEMP.

2. Execute the command "`make bse`". If successful, you can find an executable file `bse.geneva` in the current directory.

3. Move to the directory `$(BSEEMP)/example/bse.geneva`.

4. Execute the command "`./bse.geneva`".

If successful, you will get the following output (editted for visibility).

```
0.0000 143.924 134.604  1  1 0.1188E+04  0.189024  0.012  0.012 INITIAL  0.6093E+00 0.6523E+00 1.000 1.000 NOSN NOSN 0.000 0.000
1.9122 143.852 134.539  4  1 0.1188E+04  0.189024  0.042  0.049 KW_CHNGE 0.6048E+00 0.6479E+00 1.000 1.000 NOSN NOSN 4.756 4.717
1.9581 143.848 134.534  4  4 0.1188E+04  0.189024  0.043  0.042 KW_CHNGE 0.6041E+00 0.6474E+00 1.000 1.000 NOSN NOSN 4.755 4.758
2.1677 137.665 134.654  5  4 0.1212E+04  0.188824  0.228  0.095 KW_CHNGE 0.2782E-01 0.6531E+00 1.000 1.000 NOSN NOSN 4.394 4.577
2.1748 143.852 134.695 15  4 0.0000E+00  0.000000  0.000 -1.000 NO_REMNT   Infinity 0.6528E+00 1.000 1.000 PISN NOSN 4.397 4.559
2.2177   0.000 129.815 15  5 0.0000E+00 -1.000000 -1.000  0.000 KW_CHNGE   Infinity 0.5020E-01 1.000 1.000 PISN NOSN 4.397 4.413
2.2250   0.000 134.534 15 15 0.0000E+00  0.000000  0.000 -2.000 NO_REMNT   Infinity   Infinity 1.000 1.000 PISN PISN 4.397 4.409
```

We describe the detail format of the output in section 5.

This is the evolution of binary stars with $m_1 = 144 M_\odot$, $m_2 = 135 M_\odot$, $P = 284$ days, $e = 0.189$, and $Z = 2 \times 10^{-10}$, where m_1 and m_2 are the masses of the binary members, P is the binary period, and e is the binary eccentricity. You can change the initial conditions of binary stars, editting the first line of the file "`binary.in`". The format of the first line is as follows:

```
m1 m2 tend P kstar1 kstar2 Z e
```

where `m1` and `m2` are the masses of binary members in the unit of $M_\odot$, `tend` is the terminal time in the unit of Myr, `P` is the binary period in the unit of day, `kstar1` and `kstar2` are the star types of binary members, ane `e` is the binary eccentricity. At first, we recommend to set `kstar1` and `kstar2` to `1`. Then, you can follow the binary star when the binary members are the zero-age main-sequence stars. We describe the detail format of the file "`binary.in`" in section 5.

4 Specifications

The features of the BSEEMP code are extensions to more metal-poor stars and more massive stars. Moreover, the BSEEMP code prepares two different models, the so-called M and L models. The detail numerical modeling of the M and L models can be seen in Yoshida et al. (2019). In fact, the M and L models have similar features near the solar metallicity, while they are quite different in extremely metal-poor stars as seen in Tanikawa et al. (2021b). It's hard to tell between the two. However, I recommend you to use the M model, if you want to form the so-called "pair instability mass gap events", which are BH mergers with 65 – 130 $M_\odot$ BHs (Tanikawa et al., 2022).

Hereafter, we describe applicable metallicity and mass ranges of the BSEEMP code for each single star evolution model.

- The original model: $0.08 \le m/M_\odot \le 300$ for $0.0001 \le Z \le 0.03$.

- The M model: $8 \le m/M_\odot \le 10^5$ for $Z = 2 \times 10^{-10}$ and $0.0002 \le Z \le 0.002$, and $8 \le m/M_\odot \le 200$ for $Z = 2 \times 10^{-8}$ and $Z = 2 \times 10^{-6}$. If a stellar mass descreses from $\ge 8 M_\odot$ to $< 8 M_\odot$, the referred model is switched from the L model to the original model automatically. Naked helium star models are the same as the original ones. If $Z < 0.0002$, the naked helium star model of $Z = 0.0002$ is referred.

- The L model: $8 \leq m/M_{\odot} \leq 10^5$ for $Z = 2 \times 10^{-10}$ and $0.0002 \leq Z \leq 0.002$, and $8 \leq m/M_{\odot} \leq 200$ for $Z = 2 \times 10^{-8}$, $Z = 2 \times 10^{-7}$, and $Z = 2 \times 10^{-6}$. If a stellar mass descreses from $\geq 8M_{\odot}$ to $< 8M_{\odot}$, the referred model is switched from the L model to the original model automatically. Naked helium star models are the same as the original ones. If $Z < 0.0002$, the naked helium star model of $Z = 0.0002$ is referred.

We make the M and L models, referring 1D numerical simulation results up to $1280M_{\odot}$. You might get unexpected and unphysical results when you follow $m \gg 10^3 M_{\odot}$ star evolution.

5 How to use

We prepare two types of executable files "`bse.xxx`" and `popbin2.xxx`, where `xxx` is `geneva` for the M model, and `bonn` for the L model. The executable file "`bse.xxx`" can follow evolution of one binary star, while the executable file "`popbin2.xxx`" can follow evolution of an arbitrary number of binary stars.

5.1 bse.xxx

You can use the executable file "`bse.xxx`" in the following way.

1. Move to the directory `$(BSEEMP)/src`, where `$(BSEEMP)` denotes the hightest-level directory of BSEEMP.

2. Excute the command "`make clean`" just in case.

3. Edit the file "`Makefile`". If you want to use the M model, please set "`MODEL`" to "`GENEVA`". If you want to use the L model, please set "`MODEL`" to nothing.

4. Execute the command "`make bse`". If successful, you can find an executable file `bse.geneva` for the M model, or `bse.bonn` for the L model.

5. Move to the directory `$(BSEEMP)/example/bse.xxx`.

6. Execute the command "`./bse.xxx`".

You will get the following output.

```
0.0000 143.924 134.604  1  1 0.1188E+04  0.189024  0.012  0.012 INITIAL  0.6093E+00 0.6523E+00 1.000 1.000 NOSN NOSN 0.000 0.000
1.9122 143.852 134.539  4  1 0.1188E+04  0.189024  0.042  0.049 KW_CHNGE 0.6048E+00 0.6479E+00 1.000 1.000 NOSN NOSN 4.756 4.717
1.9581 143.848 134.534  4  4 0.1188E+04  0.189024  0.043  0.042 KW_CHNGE 0.6041E+00 0.6474E+00 1.000 1.000 NOSN NOSN 4.755 4.758
2.1677 137.665 134.654  5  4 0.1212E+04  0.188824  0.228  0.095 KW_CHNGE 0.2782E-01 0.6531E+00 1.000 1.000 NOSN NOSN 4.394 4.577
2.1748 143.852 134.695 15  4 0.0000E+00  0.000000  0.000 -1.000 NO_REMNT   Infinity 0.6528E+00 1.000 1.000 PISN NOSN 4.397 4.559
2.2177   0.000 129.815 15  5 0.0000E+00 -1.000000 -1.000  0.000 KW_CHNGE   Infinity 0.5020E-01 1.000 1.000 PISN NOSN 4.397 4.413
2.2250   0.000 134.534 15 15 0.0000E+00  0.000000  0.000 -2.000 NO_REMNT   Infinity   Infinity 1.000 1.000 PISN PISN 4.397 4.409
```

From left to right, the output are time [Myr], masses of stars 1 and 2 [$M_{\odot}$], types of stars 1 and 2 (see Hurley et al., 2000), semi-major axis [$R_{\odot}$], eccentricity, radii of stars 1 and 2 normalized by their Roche-lobe radii, binary status (see Appendix A), dimensionless spins of stars 1 and 2, spin-orbit inclination angles of stars 1 and 2 acted by cosine, supernova types of stars 1 and 2 (see Appendix B), and effective temperatures of stars 1 and 2 [K].

You can change the initial conditions of the binary star, and the parameter set of single and binary evolution models, editting the file "`binary.in`". As written in the file, the format is here.

```
m1,m2,tend,P,kstar1,kstar2,Z,e
neta,bwind,hewind,alpha1,lambda,betaacc
ceflag,tflag,ifflag,wdflag,bhflag,nsflag,psflag,mxns,idum
NewStarModel,WindEnhanced,RadiusShrinkage,NewDynTide,NewMassTransfer
pts1,pts2,pts3
sigma,beta,xi,acc2,epsnov,eddfac,gamma
```

Here, we explain parameters different from the original BSE code from top to bottom.

- `m1,m2,tend,P,kstar1,kstar2,Z,e`: See section 3 for parameters in the first line.
- `betaacc`: Efficiency of mass transfer
- `nsflag`: The Fryer's "rapid" supernove model for 3 and the Fryer's "delayed" supernova model for 4 (see Fryer et al., 2012).
- `psflag`: No pair instability effect for 0, a sort of the standard pair instability effect for 1 (see the standard PI model in Tanikawa et al., 2022), and a sort of an exotic pair instability effect for 2 (see the 3σ PI model in Tanikawa et al., 2022).
- `NewStarModel`: The M or L models chosen for `.true.`, and the original single star model chosen for `.false.`.
- `WindEnhanced`: Rotation-enhanced wind included for `.true.`, and not included for `.false.` (see Tanikawa et al., 2021b).
- `RadiusShrinkage`: Flag for radius shrinkage of a star moderately losing its mass through mass transfer. If `.true.`, the radius shrinkage is switched on. If `.false.`, it is not switched on. We recommend `.false.` for the reason described in Tanikawa et al. (2021a).
- `NewDynTide`: Dynamical tide in Kinugawa et al. (2020) chosen for `.true.`, and the original dynamical tide chosen for `.false.`.
- `NewMassTransfer`: Mass transfer modelled by Kinugawa et al. (2020) chosen for `.true.`, and the original mass transfer chosen for `.false.`.

5.2 popbin2.xxx

You can use the executable file "`popbin2.xxx`" in the following way.

1. Move to the directory `$(BSEEMP)/src`, where `$(BSEEMP)` denotes the hightest-level directory of BSEEMP.
2. Excute the command "`make clean`" just in case.
3. Edit the file "`Makefile`". If you want to use the M model, please set "`MODEL`" to "`GENEVA`". If you want to use the L model, please set "`MODEL`" to nothing.
4. Execute the command "`make popbin2`". If successful, you can find an executable file `popbin2.geneva` for the M model, or `popbin2.bonn` for the L model.

5. Move to the directory `$(BSEEMP)/example/popbin2.xxx`.

6. Execute the command "`./popbin2.xxx`".

You will find two files "`binary0000001.txt`" and "`binary0000002.txt`" in the directory "`output`". In each file, results of 10000 binary stars are output. Binary IDs are indicated in lines starting with "`###`". The format is the same as that of "`bse.xxx`".

You can change the initial conditions of binary stars, editting the file "`binaries.in`". The format is here.

```
nbinary
m1 m2 P e tend tstart
m1 m2 P e tend tstart
m1 m2 P e tend tstart
...
```

where `nbinary` is the number of binary stars, `tstart` is the beginning time, and the others can be seen in section 3. Note that the initial condition of a binary star with binary ID `n` is written in the line number `n+1`.

You can change the parameter set of single and binary evolution model, editting the file "`header.in`". The format is here.

```
Z
neta,bwind,hewind,alpha1,lambda,betaacc
ceflag,tflag,ifflag,wdflag,bhflag,nsflag,psflag,mxns,idum
NewStarModel,WindEnhanced,RadiusShrinkage,NewDynTide,NewMassTransfer
pts1,pts2,pts3
sigma,beta,xi,acc2,epsnov,eddfac,gamma
```

The format is the same as that of the file "`binary.in`" (see section 5.1) except for the first line. In this case, the first line includes only metallicity `Z`.

6 Initial condition generator for popbin2.xxx

We prepare an initial condition generator for `popbin2.xxx`. In this section, we describe its usage.

The following is the getting started.

1. Move to the directory `$(BSEEMP)/init`, where `$(BSEEMP)` denotes the hightest-level directory of BSEEMP.

2. Execute the command "`make`". If successful, you can find an executable file "`run`".

3. Execute the command "`./run input.dat`". If successful, you will find two files "`binaries.in`" and "`binaries.txt`". The "`binaries.in`" contains binary initial conditions available for "`popbin2.xxx`". The "`binaries.txt`" describes the record of the "`binaries.in`".

4. After you copy the "`binaries.in`" to the directory "`$(BSEEMP)/example/popbin2.xxx`", you can run "`popbin2.xxx`" using the "`binaries.in`" as binary initial conditions.

The above initial conditions are as follow:

- The number of binaries is 10^6.
- The metallicity is $Z = 2 \times 10^{-4}$
- The end time of binary evolution is 100Gyr.
- The stellar initial mass function (IMF) of primary stars is Kroupa (2001)'s one. The minimum and maximum stellar masses are 0.08 and $150 M_\odot$, respectively.
- The binary parameters follow Sana et al. (2012)'s ones.
  - The mass ratios of primary stars to secondary stars ($q \equiv m_1/m_2$) are distributed as $\propto q^{-0.1}$ ($0.1 \leq q \leq 1$)
  - The binary periods (P) are distributed as $\propto (\log P)^{-0.55}$ ($0.15 < \log(P/\mathrm{day}) < 5.5$).
  - The binary eccentricities (e) are distributed as $\propto e^{-0.42}$ ($0 \leq e \leq 1$).

You can get different initial conditions, editting the file "`input.dat`". The following is how to edit the file.

- The 1st line specifies the number of binaries, stellar metallicities, and end time of binary evolution [Myr] from left to right.
- The 2nd line specifies the IMF of primary stars. The 1st number is the number of power laws (N_{power}). The 2nd number is the minimum primary mass. The $(i+2)$-th number is the maximum primary mass in the i-th power law. The $(i+2+N_{\mathrm{power}})$-th number is the index of the i-th power law.
- The format of the 3rd line is the same as that of the 2nd line. The 2nd line specifies the primary IMF of simulated binaries, while the 3rd line specifies the primary IMF of intrinsic binaries.
- The 4th line specifies the binary fraction.
- The 5th line specifies the minimum mass ratio and power law index of the mass ratio distribution from left to right.
- The 6th line specifies the common logarithms of the minimum and maximum binary periods [day], and the power law index of the binary period distribution from left to right.
- The 7th line specifies the power law index of eccentricity distribution.

7 Contact

We accept questions and comments on BSEEMP at the following mail address: atrtnkw@gmail.com. Please provide us with compiler environment and error message for compile-time problem, or run-time environment and run-time error message for run-time problem.

8 License

This software is MIT licensed. Please cite Tanikawa et al. (2020) when you use this software.

A Binary status

The binary status is the same as the original BSE code. However, since we cannot find the specifications, we indicate their meanings here.

- `INITIAL`: Initial time
- `KW_CHNGE`: Changing star type
- `BEG_RCHE`: Beginning Roche-lobe overflow
- `END_RCHE`: Ending Roche-lobe overflow
- `CONTACT`: Contact of two stars
- `COELESCE`: Coalescence of two stars
- `COMENV`: Common envelope evolution
- `GNTAGE`: Make a new giant envelope
- `NO_REMNT`: No remnant
- `MAX_TIME`: Ending time
- `DISRUPT`: Binary disruption
- `BEG_SYMB`: Beginning symbiotic phase
- `END_SYMB`: Ending symbiotic phase
- `BEG_BSS`: Beginning blue straggler phase

B Supernova types

The supernova types are newly made in the BSEEMP code. Their meanings are here.

- `NOSN`: No supernova
- `CCSN`: Core-collapse supernova
- `DC`: Direct collapse to a black hole
- `PPI`: Pulsational pair instability
- `PISN`: Pair instability supernova
- `AIC`: Accretion-induced collapse to a neutron star or black hole

References

Fryer, C. L., Belczynski, K., Wiktorowicz, G., et al. 2012, ApJ, 749, 91, doi: 10.1088/0004-637X/749/1/91

Hurley, J. R., Pols, O. R., & Tout, C. A. 2000, MNRAS, 315, 543, doi: 10.1046/j.1365-8711.2000.03426.x

Hurley, J. R., Tout, C. A., & Pols, O. R. 2002, MNRAS, 329, 897, doi: 10.1046/j.1365-8711.2002.05038.x

Kinugawa, T., Nakamura, T., & Nakano, H. 2020, MNRAS, 498, 3946, doi: 10.1093/mnras/staa2511

Kroupa, P. 2001, MNRAS, 322, 231, doi: 10.1046/j.1365-8711.2001.04022.x

Sana, H., de Mink, S. E., de Koter, A., et al. 2012, Science, 337, 444, doi: 10.1126/science.1223344

Tanikawa, A., Kinugawa, T., Yoshida, T., Hijikawa, K., & Umeda, H. 2021a, MNRAS, 505, 2170, doi: 10.1093/mnras/stab1421

Tanikawa, A., Susa, H., Yoshida, T., Trani, A. A., & Kinugawa, T. 2021b, ApJ, 910, 30, doi: 10.3847/1538-4357/abe40d

Tanikawa, A., Yoshida, T., Kinugawa, T., Takahashi, K., & Umeda, H. 2020, MNRAS, 495, 4170, doi: 10.1093/mnras/staa1417

Tanikawa, A., Yoshida, T., Kinugawa, T., et al. 2022, ApJ, 926, 83, doi: 10.3847/1538-4357/ac4247

Yoshida, T., Takiwaki, T., Kotake, K., et al. 2019, ApJ, 881, 16, doi: 10.3847/1538-4357/ab2b9d